Guía práctica para dominar VRML 2.0

MUNDOS EN MOVIMIENTO

VRML 2.0

pcmanía

15 años

HOBBY PRESS



Edita: Hobby Press S.A.
Redacción: Ernesto Martí, Mónica Ballartín.
Edición y diseño: Equipo PCmanía.

Imprime: Pentacrom S.L.
Depósito legal M-34844-92
Este suplemento se incluye conjunta e inseparablemente con PCmanía

Presentación

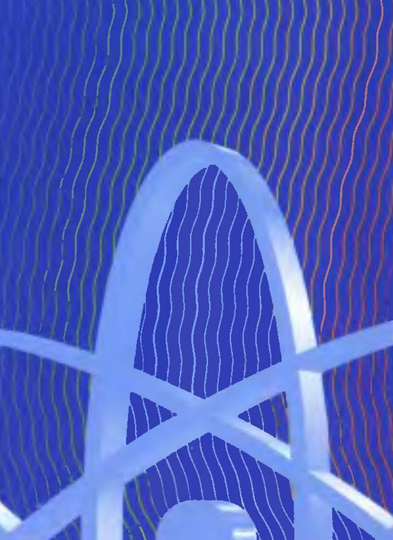
VRML



Internet, ese gran océano de información, está en constante movimiento,

las novedades se lanzan a la red ininterrumpidamente y lo que hoy es futuro. mañana es pasado. Con el lenguaje VRML. ha ocurrido una cosa similar, cuando aún nadie domina completamente la versión 1.0, ya han aparecido las especificaciones de la versión 2.0. En este libro, pretendemos acercaros al borrador 3 de las últimas especificaciones del lenguaje VRML, la que se conoce como "Moving Worlds", en una carrera por manteneros informados de las últimas noticias en el basto campo informático.

Índice



Introducción

VRML



Centrarnos en estas páginas en las especificaciones de la versión 2.0 de VRML,

que aún no utilizan todos los programas que podemos emplear para crear o visualizar mundos virtuales, puede parecer a priori una pequeña incongruencia. Sin embargo, de este modo, vosotros mismos podréis ver y comprender con facilidad las diferencias entre las dos versiones del lenguaje, y estaréis preparados para dar un nuevo paso siguiendo el ritmo que marcan los tiempos.

En el CD-ROM que acompaña a la revista, encontraréis una selección de programas, básicamente herramientas de autor, que trabajan con la versión 1.0, y en este libro podéis consultar la descripción de las especificaciones de la 2.0. La versión definitiva de VRML 2.0 en el momento de publicarse este libro debe estar disponible. En unos días, estaréis preparados para acometer el trabajo de crear realidad virtual y distribuirla por la red con seguridad, sin tener que emplear un tiempo extra para comprender las modificaciones y extensiones respecto a la versión 1.0. Además, contaréis con una práctica fluida de los programas que se dedican a modelar mundos virtuales, espacios virtuales o realidad virtual en la red.

La sintaxis del lenguaje se ha modificado en relación a la versión 1.0, las novedades que se han incorporado son relevantes e importantes, como veréis más adelante. "Moving Worlds", abre el camino a lo que puede llegar a ser la realidad virtual utilizada en red, espacios multidisciplinares y multiusuario, con conversaciones en vivo y en directo, utilizando nuestras tarjetas de sonido; interactividad, en una palabra, multimedia en un sentido amplio.





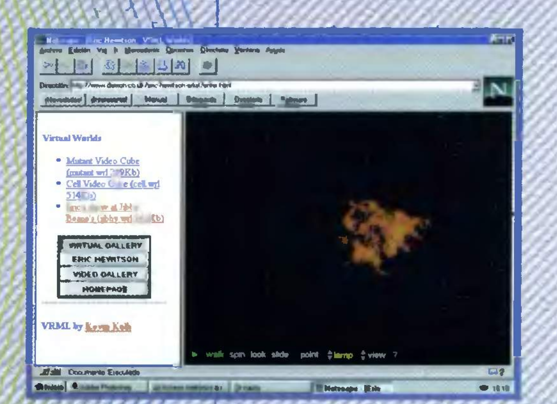


Como ha sucedido con el lenguaje HTML; VRML está abierto a nue-





vas implementaciones e incorporaciones por nuestra parte; es posible, por ejemplo, aumentar las posibilidades del lenguaje o definir alguna nueva función, teniendo en cuenta, eso sí, algunas limitaciones inherentes al sistema. El gran handicap es, sin duda, la actual velocidad en la transmisión de datos mediante la línea telefónica. Evidentemente, hoy por hoy, no es lo suficientemente amplia como para permitir un uso extensivo de este tipo de lenguaje, ¡todos sabemos lo desesperante que puede llegar a ser bajar una simple página HTML cuando todo el mundo está conectado!, pero paciencia todo llegará.



Internet, es por el momento un maravilloso lugar, en el que las novedades son continuas y en el que la tecnología tiene un fabuloso campo de cultivo.

En el desarrollo de estos lenguajes, podemos participar todos, ya sea sugiriendo ideas o bien, si nuestros conocimientos nos lo permiten, formar parte de algún equipo de desarrollo. Pero las opciones no terminan en estos puntos, si lo deseamos, podemos apuntarnos a algún foro de discusión, y recibir periódicamente noticias acerca de los diferentes estadios por los que pasa el desarrollo y cómo no, consultar todas y cada una de nuestras dudas. La soledad que impone el ordenador, no ha de ser necesariamente un enclaustramiento. Las relaciones humanas "naturales", las que tenemos habitualmente son imprescindibles y necesarias para nuestra estabilidad emocional y nuestras necesidades básicas; una máquina nunca podrá suplir esto, pero sí podemos hacer que las máquinas funcionen de una forma más humana y permitan a todas aquellas personas que, por la razón que sea, estén limitadas en algún sentido, acercarles un mundo virtual lleno de virtuosismo.







La evolución VRML



En 1994 se dió a conocer un lenguaje que revolucionó Internet

de la mano de Tim Benners Lee, el conocido HTML. En ese mismo momento, se planteó la posibilidad de crear unas nuevas especificaciones para un lenguaje que permitiera distribuir mundos generados, utilizando un lenguaje descriptivo de realidad virtual.

Durante la Primera Conferencia de la WWW se da el primer paso para el nacimiento del lenguaje que permite distribuir mediante la red espacios virtuales, y que se conoce como VRML. De este lenguaje, justo se termina de "escribir" su última revisión, la que se conoce en la red como "MOVING WORLDS", la especificación 2.0 del lenguaje VRML.

Las últimas noticias

15/7/96

•Se publica oficialmente el "Borrador #3" de la versión 2.0.

12-14/7/96

•Se determinan gran número de enlaces, se clarifican y se corrigen errores.

11/7/96

•Borrador no oficial sobre el "borrador #3".

•Se cambia la opción "wichChild" por la de "wichChoice"

•Se incluyen los caracteres "+" y "-" en el grupo de los que no se pueden utilizar como inicio de los nombres.

•Se clarifican los "Plane Sensor" por defecto para los ejes XY del plano.

•Se determinan errores de menor importancia.



9/7/96





•Se determina que: USE, PROTO, EXTERNPROTO, TRUE, FALSE, NULL, eventIn, eventOut, field, exposedField, IS, y TO son nombres no utilizables.

7/7/96

•Se realizan cambios en "SFImage" pasando a utilizar RGBA en lugar de RGBT, siendo "A" la opacidad (o alpha) y "T" la transparencia. Esto se debe a que la mayoría de los formatos de las imágenes y las librerías utilizan el primero.

3/7/96

•Se realizan cambios de tamaño en varios elementos:

-El radio por defecto de las opciones "Point-





Light" y "SpotLight" pasan a ser de 100 metros.
-El tamaño por defecto de los campos de navegación se cambian, adaptándose a medidas más comunes.





•También, se producen modificaciones que permiten cambiar la geometría desde un script.
•Se mejoran las opciones de estilo de fuente.
•Dentro de "Fog" se incluyen varias modificaciones:
-Se cambia el tamaño del campo, pasa a tener la opción de ser un nodo limitado, es decir sólo puede estar activo cada vez. De este modo, se permite una implementación más sencilla.
- También, se cambia la visibilidad por defecto de 1 a 100.
En estos cambios se incluyen los dos valores que tendrá la opción "FogType", longitudinal y exponencial.
•Se realizan modificaciones en "CylinderSensor", "PlaneSensor", y en "SphereSensor".
•Se varía el tamaño de todos los campos de las cajas, siendo (-1, -1,-1) el especificado por defecto y (0, 0, 0) el menor tamaño.
•Se determina no pluralizar los campos múltiples ("MF").

VRML

Créditos de VRML 2.0

Esta amplia lista de personajes, nos proporciona una idea del esfuerzo que ha sido necesario

hacer para mejorar la versión anterior. Los nombres de los autores y colaboradores, en sus direcciones email, nos revelan el trabajo que determinadas compañías están realizando para que VRML "Moving Worlds" sea un lenguaje con mucho futuro.



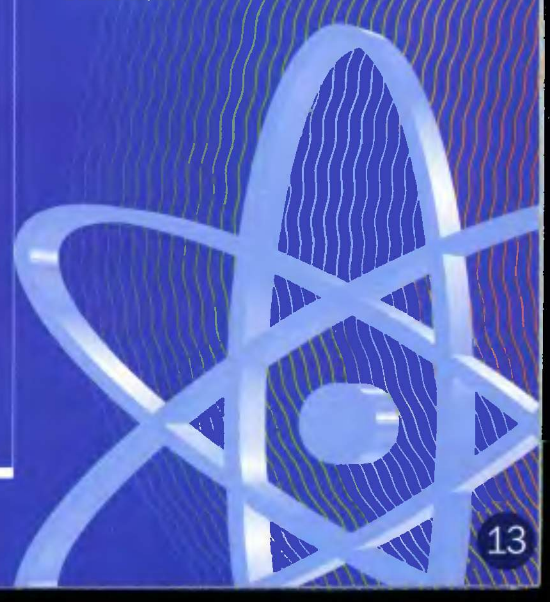
Bernie Roehl

Autores

Gavin Bell, gavin@acm.org
Rikk Carey, rikk@best .com
Chris Marrin, cmarrin@sgi.com

Contribuciones

Ed Allard, eda@sgi.com







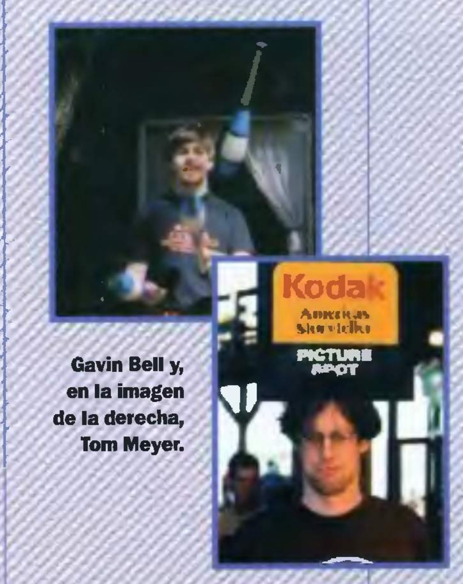
Mark Pesce, sobre estas líneas, y Don Brutzman.

Curtis Beeson, curtisb@sgi.com
Geoff Brown, gb@sgi.com
Sam T. Denton, denton@maryville.com
Christopher Fouts, fouts@atlanta.sgi.com
Rich Gossweiler, dr_rich@sgi.com
Jan Hardenbergh, jch@jch.com
Jed Hartman, jed@sgi.com
Jim Helman, jimh@sgi.com
Yasuaki Honda, honda@arch.sony.co.jp
Jim Kent, jkent@sgi.com
Rodger Lea, rodger@csl.sony.co.jp
Jeremy Leader, jeremy@worlds.net
Kouichi Matsuda, matsuda@arch.sony.co.jp
Mitra, mitra@earth.path.net
David Mott, mott@best.com
Chet Murphy, cmurphy@modelworks.com
Michael Natkin, mjn@sgi.com
Rick Pasetto, rsp@sgi.com
Bernie Roehl, broehl@sunee.uwaterloo.ca
John Rohlf, jrohlf@sgi.com
Ajay Sreekanth, ajay@cs.berkeley.edu
Paul Strauss, pss@sgi.com
Josie Wernecke, josie@sgi.com
Daniel Woods, woods@sgi.com

Revisiones

Yukio Andoh, andoh@dst.nk-exa.co.jp
Gad Barnea, barnea@easynet.fr
Philippe F. Bertrand, philippe@vizbiz.com
Don Brutzman, brutzman@cs.nps.navy.mil
Sam Chen, sambo@sgi.com
Mik Clarke, raz89@dial.pipex.com
Justin Couch, jtc@hq.adied.oz.au
Ross Finlayson, raf@tomco.net
Clay Graham, clay@sgi.com
John Gwinner, 75162.514@compuserve.com
Jeremy Leader, jeremy@worlds.net
Braden McDaniel, braden@shadow.net
Tom Meyer, tom@tom.com
Stephanus Mueller, steffel@blacksun.de
Rob Myers, rob@sgi.com
Alan Norton, norton@sgi.com
Tony Parisi, tparisi@intervista.com
Mark Pesce, mpesce@netcom.com
Scott S. Ross, ssross@fedex.com
Hugh Steele, hughs@virtuality.com
Helga Thorvaldsdottir, helga@sgi.com
Chee Yu, chee@netgravity.com

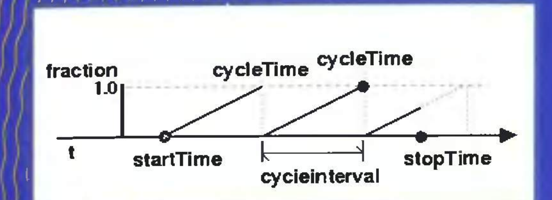
Toda la comunidad VRML, www-vrml@wired.com



Gavin Bell y, en la imagen de la derecha, Tom Meyer.



Especificaciones VRML 2.0



El lenguaje "Virtual Reality Modelling Language" (VRML) se ha desarrollado

con la finalidad de permitir generar mundos virtuales y objetos en 3D con capacidades interactivas.

A pesar del creciente número de espacios virtuales en Internet, el lenguaje HTML continúa siendo el más común. Sin pretender usurpar su puesto, los diseñadores de esta nueva versión del lenguaje VRML, han pretendido sin duda hacerlo mucho más asequible a los usuarios.

Estos espacios virtuales creados mediante VRML juegan la importante baza de generarse para la WWW, con la opción de constituir mundos multiusuario. De hecho, ésta es una de las principales orientaciones que se está dando a este lenguaje, la de acabar compartiendo mundos multiusuario interactivos.

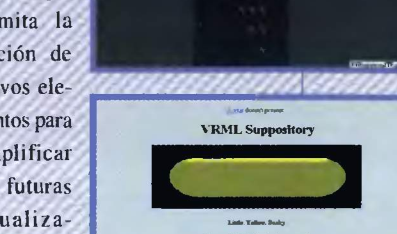
Entre las múltiples funciones que se le atribuyen están las de crear representaciones en 3D de escenas complejas, definiciones de productos y presentaciones de realidad virtual.

El lenguaje VRML ha sido diseñado persiguiendo unos criterios que hagan de él:

•Un lenguaje de autor que permita importar ficheros con otras extensiones distintas a las suyas y desarrollar aplicaciones. De este modo, el tiempo de creación de los objetos se reducirá y se hará más accesible.

•Un lenguaje que provea de toda la información necesaria para implementar y determinar un desarrollo completo que resulte competitivo.

•Un lenguaje reciclable que permita usar elementos de VRML combinados permitiendo reutilizarlos.



•Un lenguaje extensible, esto es, que permita la adición de nuevos elementos para simplificar las futuras actualizaciones.

En este mismo sentido se ha pretendido potenciar sus características de multiusuario, que en un futuro no muy lejano (y siempre que se incremente la velocidad de las líneas telefónicas) permita los entornos compartidos.

-Un lenguaje multiplataforma, permitiendo la implementación en un amplio margen de sistemas.

•Un lenguaje donde los elementos que lo constituyen deben ser independientes unos de otros, o que las interrelaciones que existan entre ellos se encuentren perfectamente determinadas y estructuradas.







De este modo se evitarán los conflictos y se conseguirá un lenguaje mucho más funcional.

•Un lenguaje potente; sus elementos están diseñados con el fin de permitir su ejecución interactiva en una gran variedad de plataformas.

•Un lenguaje escalable, los elementos del lenguaje están pensados para poder ser incluidos en espacios (potencialmente) infinitos, sólo determinados por la imaginación del autor ;-))

•Un lenguaje con estándares adaptados a la realidad de la práctica. Se pretende evitar la implantación de un gran número de elementos estándares que dificulten la utilización del lenguaje. Solo aquellos elementos que sean prácticos, que sean necesarios para soportar las prácticas existentes o para soportar estándares propuestos deberían ser estandarizados. Con este lenguaje, se pretende conseguir un entorno de usuario muy definido, con una estructura clara. Se han evitado los elementos con varios propósitos y con distintos efectos.

Características del lenguaje

El lenguaje VRML nos permite la descripción de escenas constituidas por elementos en 3D, que pueden estar en movimiento o ser estáticos. Estas escenas pueden, al igual que las páginas HTML, contener enlaces con otros puntos de la red, ya sean otros mundos virtuales o páginas 2D. Entre sus capacidades multimedia se encuentran las de enlazar con imágenes, sonidos y vídeo.

Actualmente existen navegadores de VRML para la mayoría de plataformas del mercado, y una amplia variedad de herramientas de autor. Aunque no sea necesario saber el lenguaje en sí, de hecho los modeladores son en su mayoría 100% visuales, si es interesante conocer un poco su estructura y funcionamiento para descubrir alguna característica que el modelador no contemple, y aumentar la calidad de nuestros trabajos.

La versión 2.0 del lenguaje VRML, Moving Worlds, permite un mayor número de opciones que su predecesora, uno de sus principales éxitos consiste en haber conseguido un lenguaje que ofrece al usuario una navegación realmente interactiva y crear la base sobre la que desarrollar posteriores ampliaciones del VRML.

Algunas de las principales características que se han mejorado en la versión VRML 2.0 son:

• Mundos más reales:

Gracias a las nuevas características del lenguaje, podremos aumentar el realismo a los mundos que hemos creado. Los nuevos nodos nos permiten crear escenas con paisajes en la distancia, objetos con niebla. También se han incorporado otros que nos permitirán crear superficies irregulares, en lugar de ser totalmente planas.



En este mismo sentido, la posibilidad de incluir "nodos" de sonido espaciales aumenta el realismo de las





escenas creadas. Incluso las modificaciones se han pensado de tal modo que la creación de navegadores para VRML -simplificando las tareas de optimizar ficheros- sea relativamente sencillo gracias a que utiliza una estructura gráfica simplificada.

•Interactividad:

Gracias a los nuevos nodos sensores, podremos añadir eventos a determinadas áreas o a determinados objetos. Podremos determinar tiempos de repetición para las animaciones, desplazar objetos o controles de un lugar a otro.

Se acabó atravesar paredes u objetos, confundiéndonos. La implantación de detectores de colisión permite que los objetos sólidos actúen como tales, así como en la vida real, deberemos rodearlos o detenernos, ya no atravesaremos las paredes como almas en pena como sucedía con la versión uno.

•Animación:

La animación en VRML 2.0 se realiza a partir de los denominados "Interpolators"; éstos permiten crear animaciones predefinidas de varios aspectos del mundo.

Mediante los interpoladores podremos animar todo tipo de elementos desde una mosca que vuela, hasta ventanas que se abren y nos muestran una nube que se mueve.

Otra de las opciones de animación que podremos programar es una visita guiada que mueve al usuario hacia un lugar que hemos predeterminado, en un circuito determinado por nosotros.

• Scripts:

Para asignar efectos suplementarios a las animaciones, contamos con los nodos de "Script". Estos efectos se consiguen mediante eventos: el script recoge información de los sensores, y genera acontecimientos basándose en ellos; estos eventos, a su vez, variarán determinados nodos del mundo virtual. El sistema utilizado por los eventos ("events") para pasar a través de los nodos se conoce como "routes" (carreteras o caminos).

• Creación de prototipos:

Con la nueva versión del lenguaje VRML, podremos encapsular un grupo de nodos para reunirlos en un nuevo tipo de nodo, un prototipo; para variar sus características sólo deberemos variar los valores de los distintos campos. Incluso podremos hacerlo accesible al resto de usuarios.

La sintaxis

Para comenzar a trabajar con este lenguaje, necesitamos conocer unos conceptos básicos que nos permitan comprender cómo funciona y cuál es su estructura. Para empezar deberemos identificar el fichero mediante la versión del lenguaje que estamos utilizando, y el "draft" (si estamos trabajando con un borrador, versión completa pero no definitiva). Cualquier otro caracter que incluyamos en esta línea de identificación es ignorado.

#VRML V2.0 utf8

Las últimas siglas que acompañan a la versión permiten que los caracteres internacionales sean visualizados mediante el sistema de codificación UTF-8, correspondientes al standard ISO 10646.

El caracter # empieza un comentario.



Los espacios en blanco (ya sean uno o más) separan las entidades sintác-





ticas de los ficheros VRML.
Las comas, los tabuladores, o el cambio de línea actúan como espacios en blanco donde quiera que aparezcan fuera de los campos.
Se ha de tener en cuenta que en determinados servidores de documentos VRML, pueden eliminar los espacios en blanco y los comentarios de esta parte para enviar la información con mayor velocidad. Por lo tanto la información que nos interesa preservar (copyright, creador,...) será más conveniente introducirla en el nodo "WorldInfo".
Después del encabezamiento necesario para describir a los navegadores qué tipo de fichero han solicitado (sería algo así como la tarjeta de presentación), un fichero VRML puede contener los siguientes datos:
-Prototipos definidos utilizando "PROTO" o "EXTERNPROTO".
-Nodos.
-"Routes" establecidas entre nodos.
Campo, evento y prototipo son nombres de elementos que no deben comenzar con un dígito (0x30-0x39) pero que pueden contener otro tipo de caracteres excepto los no imprimibles, comillas, #, el signo + o el -, el período ".", los corchetes, el backslash "\" o las llaves "{}".
Los caracteres en los nombres están especificados en ISO 10646 y codificados utilizando UTF-8. Hemos de tener en cuenta a la hora de editar el código que en VRML se diferencian las mayúsculas de las minúsculas.

Conceptos básicos

NODO

El lenguaje VRML define un conjunto de objetos útiles para realizar gráficos en 3D y construir mundos. Estos objetos son los nodos, conteniendo información que es almacenada en "campos" (Fields).
•Características principales:
-Tipo de objeto: esfera, textura, sonido, rotación,...
-Cuentan con un nombre que los identifica: esfera, cubo, ... Esto es útil a la hora de trabajar con ellos.
Puede ocurrir que varios nodos tengan el mismo nombre, pero no es obligatorio asignarles uno, si lo hacemos, sólo pueden tener uno.

FIELD

Existen parámetros que los diferencian de otros nodos del mismo tipo: tamaño, color, luz,... Estos parámetros distintivos son los campos ("Fields").
Un nodo pueden contar con un número de campos que vaya desde 0 a cualquier número.
Existen dos clases de campos, los que contienen un sólo valor y los que contienen varios. Los valores pueden ser un número, un vector, imágenes, ...
En caso de que el campo contenga un sólo valor, su nombre comenzará por "SF", y si tiene varios por "MF".
La sintaxis es la siguiente:
-Campos con varios valores, se representarán separados por comas y dentro de corchetes: [1,8,3]
-Campos sin ningún valor especificado, solamente corchetes: []
-Campos con un valor, se permite omitir los corchetes.
Cada especificación del nodo define el tipo, nombre y el valor por defecto para cada uno de los campos (éste aparecerá en caso de que no



haya sido determinado ninguno). El orden en que se lean los nodos no



tiene mayor importancia porque son características que afectan en conjunto. "Cone { bottomRadius 1 height 6 }" es lo mismo que "Cone { height 6 bottomRadius 1}" . En este caso, el nombre del nodo es "Cone", y los campos que lo distinguen de otros nodos que pudieran llamarse igual son radio1 y altura 6.

CHILD NODES

En VRML se ha implementado una jerarquía que permite que determinados nodos contengan a otros. De este modo, los "Parent nodes" o "Group nodes" comprenden un número determinado de nodos (que va de 0 a infinito) a los que afectarán durante el renderizado.

La sintaxis se correspondería con:

DEF objectname objecttype { fields children }

Siendo imprescindible el tipo de objeto y las llaves.



EVENTO

Los nodos pueden recibir un número de eventos que se anotan como "eventIn" y normalmente afectan al nodo. Estos pueden ser: determinar color (por ejemplo set_color), posición, ...

Los nodos pueden también enviar un número de eventos que indican que algo ha cambiado en el nodo (posición, color, ...) y que se representan como "eventOut" (por ejemplo color_changed).

La etiqueta "exposedField" puede utilizarse para especificar que un nodo ha determinado un "eventIn" que está directamente relacionado a un valor de campo y que ha cambiado a "event_Out".

Por ejemplo, la expresión "exposedField foo" es equivalente a :

eventIn set_XXX

field XXX

eventOut XXX_changed

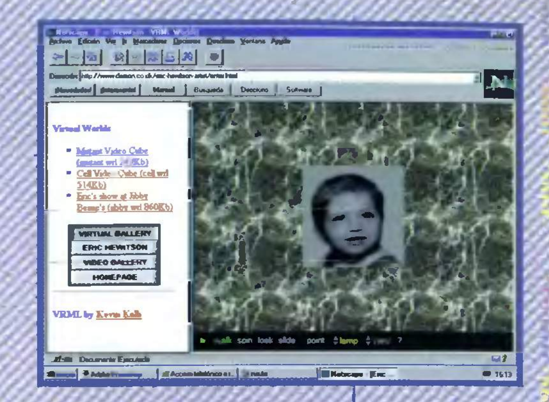
Donde "set_XXX" automáticamente determina el valor del campo "XXX" y genera un "eventOut.

La mayoría de los nodos tienen al menos un "eventIn" y de este modo pueden recibir eventos.

Los eventos que llegan son mensajes de información que son enviados por otros nodos para cambiar algún estado del receptor. Algunos nodos tienen también definiciones de "eventOut", que son utilizadas para enviar mensajes a otros nodos o para alertar a los nodos controladores que algún estado ha cambiado.

RUTAS

La conexión entre el nodo que ha generado el evento y el que lo recibe, es llamada ruta. Un nodo que produce eventos de algún tipo puede ser enviado a otro nodo que recibe eventos del mismo tipo con la siguiente sintaxis:



ROUTE "Nombre del nodo".eventOutXXX_changed TO "Nombre del nodo.set_eventInYYY

El prefijo "set_ "y el sufijo "_changed" son determinados por acuerdo, no por reglas estrictas.

Cuando se crean prototipos o scripts, los nombres de los "eventIn" y de los "eventOut" pueden ser cualquier nombre identificador (de los que están aceptados).





Existen tres excepciones a esta recomendación:

-"Bool events", eventos boleanos. Los eventIn y eventOut son conocidos como isXXX, por ejemplo isActive.

-"Time events", eventos de tiempo. En estos casos se conocen como XXXTime, por ejemplo enterTime.

-"Children events", en este caso los eventIn son sustituidos por "add" (añadir) y "remove" (eliminar).

Las vías o rutas ("routes") no son nodos, es simplemente una construcción sintáctica que se utiliza para establecer caminos entre los nodos.

Como hemos comentado, las rutas se establecen sólo desde eventOuts a eventIn, es decir desde el nodo que envía hacia el que recibe. En caso de que un fichero repita un camino, todos aquellos que lo repitan son ignorados.

Ejemplo:

DEF CLICKER TouchSensor { enabled TRUE }

DEF LIGHT DirectionalLight { on FALSE }

ROUTE CLICKER.enabled TO LIGHT.on

o

ROUTE CLICKER.enabled_changed TO LIGHT.on

o

ROUTE CLICKER.enabled TO LIGHT.set_on

o

ROUTE CLICKER.enabled_changed TO LIGHT.set_on

SENSORES

Los nodos sensores ("sensors") generan eventos, estos pueden ser de visibilidad, de proximidad, de colisión. Estos eventos están basados en las acciones del usuario, es decir generan la interactividad del espacio virtual.

PROTOTIPOS

Este es un sistema que permite la determinación de tipos de nodos para ser extendidos. Ofrece la posibilidad de ensapsularlos y dar parámetros de geometría, caracteres o ambos.

NUEVOS TIPOS DE NODOS EN LA VERSIÓN 2.0

• Grouping Nodes

-El nodo de colisión ("Collision") determina qué objetos de la escena pueden ser atravesados o no.

-Transform, agrupa los nodos juntos en una sola coordenada de sistema o "frame of reference", que incorpora los campos del antiguo nodo "Separator node".

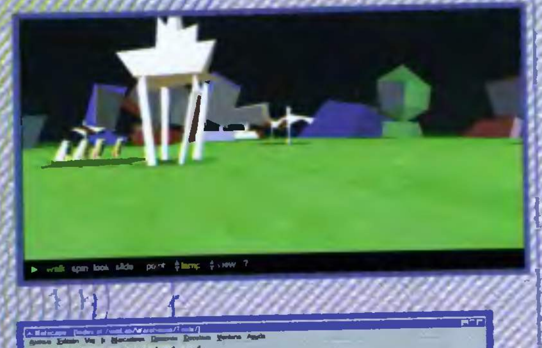




•Browser Information

Sustituyendo al nodo "Info", la nue-





va versión de este lenguaje presenta varios nuevos tipos de nodo para proveer información específica sobre la escena al navegador.

-NavigationInfo, informa al navegador sobre qué tipo de visualizador ha de utilizar en función de la velocidad de movimiento, el radio de cámara para el detector de colisión.

-Viewpoint, sustituye al nodo de "Perspective Camera". Especifica las coordenadas desde las que se visualizará la escena.

-WorldInfo, contiene la información sobre el autor de la escena, copyright.

•Lights and Lighting

-Fog, contiene la información sobre la iluminación de la escena.

•Sonido

-Define una fuente de sonido

•Forma

-Shape, especifica los nodos geométricos y las propiedades que se le aplican a ese nodo.

•Geometría

-ElevationGrid, permite especificar una superficie irregular.

-Extrusion, permite aplicar "extrude" a las formas.

-Text, reemplaza a el nodo AsciiText que aparecía en la versión anterior, simplificando su utilización.

-Propiedades geométricas.

-Color, determina colores RGB para utilizar en los campos.

•Apariencia

Appearance, reune todas las propiedades de apariencia para un nodo de forma (Shape).

•Sensores

-Sensor de proximidad ("Proximity-sensor"), genera eventos cuando la cámara se mueve en unas determinada posición y condiciones.

-Sensor de contacto ("TouchSensor"), genera eventos cuando el usuario presiona un determinado lugar.

-Existen determinados sensores para generar eventos cuando el usuario presiona en objetos como puede ser un cilindro (CylinderSensor), un plano(PlaneSensor), una esfera (SphereSensor).

-Sensor de visibilidad, genera eventos cuando una zona entra o sale del campo de vista.

-Sensor de tiempo, determina la periodicidad o el momento en que ocurren los eventos.

•Script

-Procesa los eventos.

•Interpolator

-Son utilizados para generar las animaciones, entre las novedades encontramos los referentes a color, coordinadas, orientación, posición.







Ejemplo práctico



En el CD-ROM de portada encontraréis una serie de programas que os servirán

para empezar a familiarizaros con este lenguaje, y os ayudarán a crear y visualizar correctamente vuestras propias escenas de realidad virtual para distribuirlas después por la Red.

A modo de pequeño tutorial, os planteamos el ejemplo que encontraréis en este capítulo. Para poder visualizarlo, deberéis abrir un editor de texto (el Write para Windows 3.x o el Wodpad para Windows 95 os servirán perfectamente) copiar el texto del ejemplo y salvar el fichero así generado con la extensión ".WRL" en relación con la palabra "world". De esta forma tan sencilla, y utilizando un Plug-in de VRML para vuestro browser favorito, podréis ver esta escena de realidad virtual.

El ejemplo lo hemos incluido impreso para que podáis compararlo con la sección que trata sobre la versión 2.0 del VRML. Es una forma sencilla de ver cómo funciona este lenguaje respecto al uso de "nodes" y demás componentes. Es muy práctico ver el código generado por una herramienta de autor, y compararlo con las especificaciones, ya que de este modo, y aunque rara vez tengamos que teclear código, podremos tener una idea precisa de cómo estructura el lenguaje la escena que estemos generando y, en última instancia, siempre seremos capaces de modificar una escena aunque no tengamos a mano un editor visual. ;-)

La escena que hemos generado, es un mundo al que hemos llamado PCMANÍA, y como veréis es sencillo pero ilustra todas las capacidades del lenguaje VRML.

```
#VRML V1.0 ascii
Separator {
    PerspectiveCamera {
        position      0 30 240
        orientation   0 0 -1 0
        focalDistance  5
        heightAngle   .785
    } #PerspectiveCamera
    Info {
        string "PC MANIA"
    } #Info

    Material {
        diffuseColor 1 0 1
        shininess 0.2
        transparency 0
    } #Material

    MaterialBinding {
        value  DEFAULT
    } #MaterialBinding


    FontStyle {
        size   15
        family  TYPEWRITER
        style  NONE
    } #FontStyle
```





```
AsciiText {
    string "PC MANIA"
    spacing 1
    justification CENTER
    width 0
} #AsciiText

DEF Cone1 Separator {
    Translation {
        translation 0 30 0
    } #Translation

    Cone {
        parts  ALL
        bottomRadius   15
        height       30
    } #Cone
} #Cone1 Separator

DEF Cube1 Separator {

    Transform {
        rotation 0 1 0 .7
    } #Transform

    Translation {
        translation -45 30 0
    } #Translation

    Cube {
        width   30
        height  30
        depth   30
    } #Cube

} #Cube1 Separator

DEF Cylinder1 Separator {

    Translation {
        translation 45 30 0
    } #Translation

    Cylinder {
        parts ALL
        radius 15
        height 30
    } #Cylinder

} #Cylinder1 Separator

DEF Sphere1 Separator {

    Texture2 {
        filename "wood4.jpg"
    }
    Translation {
        translation 0 75 0
    } #Translation

    Sphere {
        radius 15
    }#Sphere

} #Sphere1 Separator

DEF FaceDiamond Separator
{

        DEF DiamondCoords
Coordinate3 {
            point [
            0 75 25,
            12.5 62.5 12.5,
            12.5 62.5 37.5,
            -12.5 62.5 37.5,
            -12.5 62.5 12.5,
            0 50 25,
            ]
        } #DiamondCoords Coor-
dinate3



        USE DiamondCoords

        IndexedFaceSet {
            coordIndex [
                0, 1, 2, -1,
                0, 1, 4, -1,
                0, 4, 3, -1,
                0, 3, 2, -1,
                5, 1, 2, -1,
                5, 1, 4, -1,
                5, 4, 3, -1,
                5, 3, 2, -1,
                ]

        } #IndexedFaceSet

    } #FaceDiamond Separator

    DEF LineDiamond Separator {

        Translation {
            translation 0 75 0
        } #Translation
        USE DiamondCoords

        IndexedLineSet {
            coordIndex [
                0, 1, 2, -1,
                0, 1, 4, -1,
                0, 4, 3, -1,
                0, 3, 2, -1,
                5, 1, 2, -1,
                5, 1, 4, -1,
                5, 4, 3, -1,
                5, 3, 2, -1,
                ]

        } #IndexedLineSet
    } #DiamondLine Separator
} #Separator
```



El software

VRML



En este capítulo, veremos una importante lista de diferentes programas

que utilizaremos tanto para crear como para visualizar espacios generados con VRML y software especializado en "servir" los mundos generados en la Red.

Servidores

Existen programas especializados en facilitar páginas generadas en VRML. Este tipo de servidores, funcionan especificamente para VRML optimizando una serie de funciones para obtener mayor velocidad y comodidad en la gestión de espacios virtuales. Entre los más destacados, podemos citar:



• V-Real 3D Media Server

Se trata de uno de los programas más avanzados en este campo, con unas prestaciones muy altas.

Entre sus características más interesantes, podemos encontrar algunas que destacamos a continuación:

-Capacidad Multi-Usuario.

-Varios mundos, en un mismo servidor.

-Interacción en tiempo real.

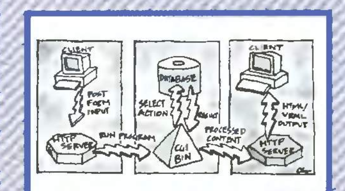
-Soporte para IRC Chat.

-Soporte para Audio Chat con LIVE AUDIO.

El programa funciona bajo Windows NT con un mínimo de 16 Mb de memoria.

• VRServer

Posee unas prestaciones ligeramente inferiores al programa anterior. Su funcionamiento se basa también en Windows NT.



Navegadores

Para poder visualizar un espacio virtual se necesita algún programa que esté preparado para ello. Esta aparente perogrullada, para usuarios con cierta experiencia, puede facilitar bastante las cosas a aquellos que están empezando a tener relaciones con la red de redes (a nosotros nos costó bastante encontrar un buen visualizador por no decir de un buen editor, en nuestros inicios).



•Cosmo Player

Cosmo Player de Silicon Graphics: es el navegador desarrollado por Silicon Graphics la empresa que, entre otras cosas, desarrolló el borrador a partir del cual se trabajó hasta conseguir el VRML 2.0. Permite sensores, scripts y sonido. La tecnología de SGI nos permitirá disfrutar de una gran experiencia. Está disponible pa-



ra Windows 95, Windows NT y por supuesto Irix:
http://www.sgi.com

•Futuresplash

Con este plug-in de la empresa FutureWave podremos disfrutar de animaciones vectoriales en tiempo real, y según la información del programa incluso a 14.400Kbps. También, permite botones interactivos, gráficos con anti-aliased.
Está disponible para Windows 95, Windows NT, Windows 3.11 en:
http://www.futurewave.com

•Iconauthor

Esta herramienta de la empresa Aimtech, permite construir aplicaciones multimedia utilizando su diseño orientado al objeto y el entorno de desarrollo visual. Sin necesidad de scripts.
Para Windows 95 y Windows 3.1 en:
http://www.aimtech.com

•MBED

Esta herramienta, puede utilizarse sin necesidad de coste, completamente gratis. Interactiva y multimedia.
Está disponible para Windows 95, Windows NT, Windows 3.1 en:
http://www.mbed.com

•Narrative Enliven Viewer

Desarrollado por la empresa Narrative Communications Corp., constituye un producto sencillo e interactivo. La versión actual es la beta1, estando pensada para conexiones a Internet de baja banda. En sus páginas web encontraremos "demos" y más información. Está disponible para Windows 95 y Windows NT en:
http://www.narrative.com

•Play3D

Ese producto de la empresa Headsoff permite entre otras opciones enlaces entre mundos en 3D, sonido, etc.
En sus páginas web encontraremos una versión gratuita.
Está disponible para Windows 95 en:
http://www.headsoff.com

•Vrealm

Este plug-in para Netscape desarrollado por la empresa INTEGRATED DATA SYSTEMS, soporta completamente VRML. Disponible para Windows 95 y Windows NT en:
http://www.ids-net.com

•Sizzler

Este producto de la empresa TOTALLY HIP, permite visualizar e interactuar con las páginas, reduciendo el tiempo de espera.
Está disponible para Windows 95, Windows 3.1 y Windows NT.

•Topper

Topper es un plug-in para Netscape 2.0 de la empresa KINETIX para la visualización de mundos en 3 D.
Soporta VRBL (Virtual Reality Behavior Language), formatos 3DS y DXF.. Disponible para Windows 95 y Windows NT en:
http://www.kts.com

•Viscape

Este plug-in de la empresa SUPERSCAPE permite navegación interactiva. Es compatible con ordenadores 486 o superior.
Desde sus páginas podremos visitar la SuperCity en realidad virtual.
Disponible para Windows 95 y Windows NT en:
http://www.superscape.com



•VR Scout Vrml

Plug-in de VRML 1.0 de la empresa Chaco. Disponible para Windows 95, Windows NT y Windows 3.1 en:
http://www.chaco.com



•Webxpresso

WebXpresso es un plug-in para Netscape Navigator que soporta la interacción en tiempo real, gráficos en 2D y 3D.
Navegador y editor.
Disponible para Windows 95, Windows NT en:
http://www.dvcorp.com

•Wirl Virtual Reality Browser

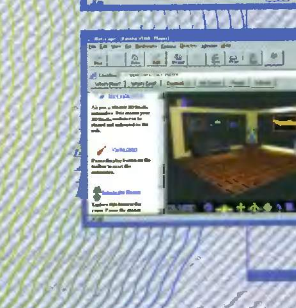
WIRL de la empresa VREAM nos ofrece una visualización excepcional bajo Windows 95 y Windows NT.
Podréis encontrar una versión de este programa en el CD-ROM que se entrega con la revista. De todos modos, sus páginas web se encuentran en:
http://www.vream.com

•Live3D

Este plug-in de Netscape, permite navegar a través de mundos virtuales de la WWW y visualizar aplicaciones multiusuario programadas en Java.

Netscape ha desarrollado los plug-in para VRML. El rápido avance de Internet le obliga a adaptarse.

Con esta nueva implementación de Netscape podremos navegar a través de mundos virtuales interactivos sin perdernos detalle puesto que permite visionar imágenes, textos en 3D, fondos con imágenes, texturas animadas... Cuenta con detector de colisiones, efecto de gravedad, y sonido Real Audio.



Para acceder a uno de estos espacios sólo deberemos presionar sobre el enlace, como en los enlaces entre páginas web tradicionales, y Netscape al ponerse en contacto con el servidor para recibir la información, procesa automáticamente los datos, y al reconocer la extensión ".wrl" abre la pantalla con las herramientas de realidad virtual.

Con ellas podremos desplazarnos a través de la escena, enlazar con otros mundos, y si contamos con tarjeta de sonido en nuestro ordenador, podremos escuchar los ficheros de audio que se encuentren incorporados.

La dirección donde podremos conseguirlo es:

http://home.netscape.com/comprod/products/navigator/version_2.0/plugins/3d_animation.html

Las plataformas sobre las que puede instalarse son Windows 95, Windows 3.1 y Windows NT.

Live3D extiende las capacidades del Navigator 3.0 consiguiendo:

-Alta calidad en la visualización de VRML. Live3D permite acceder a las escenas en 3D renderizando a gran velocidad con aceleración de hardware, compresión de datos GZIP (muy útil), procesado de los fondos y render adaptado.

-Navegación. Live3D permite navegar por los espacios virtuales, con todas las opciones permitidas por los navegadores de VRML: "walk", "fly", seleccionar un punto de vista.

-Multimedia. Esta nueva implementación para Netscape está completamente integrada con las capacidades multimedia: audio, vídeo, etc.

Herramientas de autor

Como habréis podido comprobar, el lenguaje VRML nos permite crear espacios de R.V. en Internet. En la red, podemos encontrar una serie de programas que nos facilitarán la creación de éstos mundos virtuales. Así, en la siguiente dirección nos ofrecen un completo listado de estos programas:

http://www.sdsc.edu/sdsc/partners/vrml/software/modelers.html

El programa Pioneer de la empresa Caligari Corporation, es uno de los más potentes y fáciles de utilizar. Esta herramienta no sólo nos permite modelar y publicar nuestros trabajos de realidad virtual en Internet, sino que además es un buen navegador de VRML, y podemos configurarlo como una extensión de un navegador de HTML, por eso contará con un capítulo específico más adelante.

A continuación, detallamos algunos programas que nos ayudarán a construir realidad virtual. Las opciones varían mucho entre ellos y de nuestras preferencias dependerá el utilizar uno u otro.

•Clayworks

Se trata de un programa de modelado en 3D para PC. Puede salvar los objetos generados en formato VRML. Existe una versión Shareware.

•Distributed discovery

Este programa de creación de páginas Web en 3D, permite un alto grado de interactividad a los usuarios. Sus herramientas facilitan la construcción de modelos en VRML de una forma dinámica, respondiendo a las solicitudes de los programadores.

N W E S

•EZ3D VRML Author
Este avanzado programa, de momento disponible para Solaris, Silicon Graphics y en breve para Windows NT, es un programa potente, fácil de utilizar, con opciones avanzadas de modelado en 3D, aplicación de texturas y materiales, optimización de las escenas, creación automática de enlaces y raytracing. Es una bomba.

•G Web
Es un paquete completo para la creación de escenas en VRML. Tiene capacidad de modelado, e importa en los formatos más habituales de programas de 3D CAD. Actualmente, está disponible una versión beta para los siguientes sistemas operativos: Windows 3.X, Windows NT, Irix (Silicon Graphics), Sparc.

•Land Form(tm) Land Surface Modeler
A partir de ficheros DEM genera paisajes en 3D que se pueden exportar a VRML de una forma muy sencilla. Está disponible para los sistemas operativos Windows 95 y Windows NT.

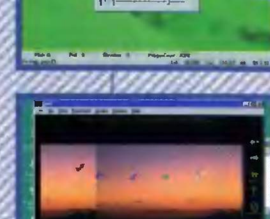
•Lightwave 3d
Este programa, uno de los mejores para modelado en 3D, nos permite salvar en formato VRML. Pensado en un principio para ordenadores Amiga, ha sido adaptado a otras plataformas. Actualmente, hay versiones para Amiga, PC y Silicon Graphics. En alguna comparativa se le ha puesto a la par con el gigante de la animación en 3D para PC, el conocido 3D Studio.

•Medit
Otro programa para plataformas Silicon Graphics. ¿Por qué será?

•Sony Ciber Passage Conductor
De la firma Sony, este programa, más bien una "suite", consta de varios módulos, entre los que se encuentran, un modelador y un navegador VRML. Utiliza una serie de extensiones VRML de la casa Sony. No es excesivamente fácil de utilizar, pero los resultados son muy buenos. Estaremos atentos a las prestaciones de las nuevas revisiones.

•Strata Studio pro Blitz
Programa para Mac con capacidades de modelado en 3D, render, animación y posibilidad de salvar en formato VRML. Incluye soporte completo para QuickDraw 3D, QuickTime VR y VRML.

•Trispectives 1.0 (v. básica y pro)
Básicamante es un programa de modelado y animación, con funciones especiales para diseñadores gráficos que quieran incorporar elementos 3D es sus creaciones. Las funciones de animación son extremadamente sencillas de utilizar, ya que podemos seleccionar un gran número de ellas, desde una simple rotación sobre cualquier eje, hasta efectos más complicados. Todo, simplemente arrastrando el efecto deseado encima del objeto. Las opciones de salvar y generar código VRML, quedan limitadas a objetos individuales. Es, sin embargo, es un programa de uso sencillo, con muchas opciones con el cual es fácil crear animaciones en 3D.



•Virtual Home Space Builder
Sencillo. Es un programa muy fácil de utilizar y consigue resultados muy interesantes. La forma en la que trabaja es bastante peculiar, puesto que construimos los mundos de una forma "cerrada" y tenemos que jugar con las escalas. Tiene una opción interesantísima que nos permite aplicar texturas animadas a los objetos.

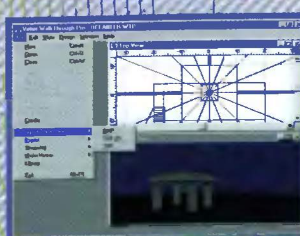
•Virtus Walkthough pro
La última versión de este conocido paquete de realidad virtual incluye la opción de salvar en formato VRML.

•Webspace author
Aunque este programa existe sólo, para Silicon Graphics esperemos que algún día pueda funcionar para ordenadores basados en procesadores Intel. Lo mencionamos aquí a











modo de homenaje, ya que es sin duda uno de los mejores programas para la creación de VRML. El programa es completamente visual y muy sencillo de utilizar. Y todo con la potencia que nos ofrecen las estaciones de trabajo de Silicon Graphics.

•Superscape VRT v.4.0
Se trata de un conocido programa profesional para la creación de realidad virtual con capacidad plena para salvar los mundos generados en formato VRML. El programa viene perfectamente documentado. Es la opción profesional para entornos DOS/WINDOWS. Consta de varios elementos perfectamente integrados: Visualizador, Creador, Editor de Imágenes ..., muy potente. Pronto disponible para Silicon Graphics.

Caligari Pioneer

Este programa de la empresa Caligari Corporation, fundada en 1986 con la finalidad de desarrollar herramientas para la creación de imágenes en 3D, nos permite navegar y crear mundos virtuales para distribuirlos en Internet. Esta compañía centra su trabajo en el difícil campo de la infografía.

Entre los productos que ofrece Caligari Corporation, el que nos ocupa en este artículo que se llama Caligari Pioneer, del que existen dos versiones, una "básica" y otra denominada Pioneer PRO, que tiene más herramientas y funciones.

Con el desarrollo y la popularización del lenguaje VRML, los navegadores para este lenguaje fueron apareciendo; en un principio se trataba de "helper applications" que venían a complementar a los navegadores tradicionales. Uno de los navegadores que aparecieron durante esta época, verano de 1995, era el "Fountain", que se trataba de la versión beta de un programa para construir VRML (al mismo tiempo que actúa como navegador) que la compañía Caligari había desarrollado.

Este programa era innovador por su sistema de trabajo, no era necesario conocer las especificaciones del lenguaje porque se trataba de un entorno completamente visual. Y Pioneer es la beta 8 del Fountain.

En el CD-ROM encontraréis Caligari Pioner, y en el web de la empresa cuya URL es http://www.caligari.com podréis acceder al resto de los programas. Por otro lado, la empresa se ha esforzado ajustando los precios de las versiones comerciales, por lo que resultan muy atractivos si nos planteamos crear espacios de R.V.

Si conocéis alguna versión del programa Caligari Truespace, no os costará adaptaros a Pioneer, tiene el mismo interfaz y los iconos realizan funciones parecidas. Para los que no habéis podido probar alguna versión de Truespace, a continuación encontraréis una aproximación al funcionamiento del programa Pioneer.

Cómo funciona Caligari Pioneer
En la creación de cualquier espacio de realidad virtual, se conjugan varios aspectos que tenemos que tener perfectamente especificados; por un lado, necesitaremos una serie de objetos en 3D, texturas, sonidos y tener muy claro los enlaces que queremos que tenga nuestro mundo.

La primera gran posibilidad del programa, es que integra funciones de modelado en 3D, nos permite importar objetos, editarlos o bien crearlos desde el programa.

Los formatos que nos permite importar el programa son:
Caligari Object, 3D Studio Binary, 3D Studio Project, 3D Studio ASCII, Autocad, Lightwave, Videoscape, Wavefront, Imagine y Caligary Amiga. Como podéis comprobar, una gran variedad que cubren perfectamente todas las opciones de importación de objetos modelados con otros programas. Pero el programa, como os hemos comentado

permite modelar, basándose en operaciones boleanas; Caligari Pioneer nos permite crear objetos en 3D ("Home World") mediante la intersección, unión, resta ..., de distintas figuras, e integrarlos en nuestra escena de realidad virtual. Pero sus funciones para el modelado no quedan ahí, tiene definidas primitivas, y podemos modelar basándonos en polígonos definibles por nosotros y después extrusionarlos para darles "cuerpo".

Uno de sus principales atractivos reside en permitir la creación de mundos virtuales a partir de un espacio en 3D, pudiendo comprobar en el mismo momento de la programación cuál va a ser el resultado final. Ya que el programa no sólo permite crear R.V. para la red, si no que es un excelente navegador de VRML (de hecho al cargar el programa, la pantalla que nos aparece, corresponde al visualizador).

Caligari Pioneer es una herramienta visual, pudiendo escoger el objeto que deseemos y situarlo sin necesidad de utilizar coordenadas. Para simplificar las tareas de crear y posicionar los objetos, incorpora un punto de vista adicional que nos permite tener dos perspectivas del elemento que estamos desarrollando. Esto es de gran utilidad ya que a veces, cuando la escena contiene muchos elementos diferentes, el posicionar exactamente un objeto donde nosotros deseamos, no es una tarea digamos sencilla. También permite dibujar polígonos en 2D y transformarlos en 3D mediante las herramientas del programa: aplicarles "extrude", levantarlos, etc.

Caligari Pioneer, por su sencillez de uso, permite que incluso los usuarios que no habían creado nunca un espacio virtual puedan aprender rápidamente a utilizarlo. Otra de las ventajas que nos ofrece con respecto a otros editores de VRML es que nos permite modificar el volumen y tamaño de los objetos fácilmente, escogiendo una de las herramientas de modificación con las que cuenta el programa.

Entre sus capacidades, se incluye la de incorporar ficheros de sonido, y graduarlos para que simulen un espacio físico, aumentando las características multimedia de los mundos. Del mismo modo, nos permite realizar todas estas operaciones en "tiempo real" y en función de la opción de trabajo que hayamos seleccionado, con objetos sombreados o sólidos.

Las fuentes True Type permiten la creación de logos y texto en 3D. A todas estas mejoras, se une la posibilidad de trabajar mediante operaciones boleanas (trabajar con varias figuras, uniéndolas, produciendo intersecciones, suprimiendo, etc.) que, por un lado, facilitan el desarrollo de objetos más realistas y, por otro, al creador de la utilización del programa.



Una vez tengamos modelados los objetos y ordenados los mundos, podemos añadirles texturas y colores que el mismo programa nos facilita; podemos asignar a los objetos atributos materiales, aplicar gradientes de colores, etc.

Dispone de la capacidad de importar/exportar ficheros, que permite (según sus autores), crear un mundo en menos de 30 minutos, utilizando "cliparts" que están disponibles por toda la red.

Los puntos de luz pueden adaptarse, de igual modo, a las características de la escena en particular, jugando con los efectos de luces y sombras.

Entre sus características más relevantes se encuentra la de incorporar "nivel de detalle"(Levels of detail o "LOD") en la navegación a través de mundos virtuales. Del mismo

modo, en aquellos mundos que creemos con esta herramienta, tendremos que tener en cuenta, que a mayor nivel de detalle, más costará en tiempo cargar el mundo cuando lo estemos visualizando en la red.
Esto significa que en función de la distancia a la que se encuentren del objeto, estos presentarán mayor o menor detalle. De este modo, cuando visualicemos la escena en perspectiva veremos más objetos, pero éstos estarán menos definidos, manteniendo la velocidad de navegación. Cuando nos aproximemos, visualizaremos menos objetos, pero con mayor precisión. Mediante esta opción, conseguiremos que aquellos usuarios que naveguen por el mundo que hemos creado, lo hagan con mayor rapidez.
Con el mismo fin, permite incorporar elementos, tales como texturas, enlaces rápidos con otros mundos, páginas HTML, compresión mediante GZIP, que reduzcan el tiempo de espera.
Los enlaces, mediante Caligari Pioneer, son sencillos. Simplemente presionando con el ratón, conseguiremos el enlace con otras páginas. Esto simplifica las tareas de creación de almacenes virtuales que puedan contener varias representaciones. En la función de navegador, podemos crear un "bookmark", pero con la particularidad de que éste será virtual. Sí, será un espacio de realidad virtual por el que podremos "pasearnos". Pulsando sobre cualquiera de los objetos que tengamos en nuestro particular bookmark en 3D, automáticamente seremos transportados a el mundo que hayamos seleccionado.
Después de una breve adaptación al programa, seréis capaces de crear vuestros propios espacios.
Una vez que habéis creado vuestro mundo virtual, sólo queda ponerlo

en la red para que los usuarios puedan navegar a través de él.
Si no disponéis de espacio en Internet, no hay problema, Caligari os ofrece la posibilidad de introducirlo en sus páginas, dentro de el "Caligari VRML Publishing World". Desde estas páginas, podréis conocer algunos de los trabajos desarrollados con esta herramienta, e incluso conseguir elementos para incorporar a vuestros mundos.
Dentro de la ayuda del programa, encontraremos varios ejemplos que nos muestran cuales son los pasos para crear un elemento en 3D con Pioneer. Cuando presionemos sobre el menú ayuda, aparecerán varias imágenes que se corresponden con los objetos cuyo desarrollo se nos muestra, el único inconveniente es que si es la primera vez que os encontráis con un programa de este tipo, deberéis visionarlo un par de veces para captar los detalles.
También, cuenta con una ayuda excepcional llamada "Help on tool" que nos suministra información sobre la herramienta que nosotros queramos.



Para empezar a familiarizarnos con este programa os recomendamos la versión básica, en ella encontraremos todas las herramientas para la construción de nuestros primeros mundos virtuales, la versión profesional del programa, está orientada a todos aquellos profesionales que por su trabajo necesitan un herramienta potente, y que reúna todas las opciones en un sólo paquete, es un buen programa para la creación de mundos profesionales.
La versión profesional del programa incorpora más capacidades. Esta herramienta está orientada a personas que quieran dedicarse profesionalmente a crear mundos o, a conseguirlos mucho más desarrollados.



Caligari Pioneer Básico

Versión actual: 1.0.

Categoría: Herramienta de autor para VRML/ VRML browser.

Sistemas operativos soportados: Windows 95, Windows NT, Windows 3.11(con Win32S).

Características del programa:

•Modelado 3D en tiempo real.

•Visualización de texturas.

•Todas las funciones en un espacio en perspectiva.

•Operaciones Boleanas en 3D.

•Sonido 3D.

•Ayuda de carácter interactivo.

•Bookmarks en formato 3D.

•Amplias opciones de importación y exportación de objetos.

•Soporte para GZIP y proxys.

Caligari Pioneer Pro

Versión actual: 1.0.

Categoría: Herramienta de autor para VRML/VRML browser.

Sistemas operativos soportados: Windows 95, Windows NT, Windows 3.11(con Win32S).

Características:

•Modelado 3D en tiempo real.

•Mapeado de texturas.

•Todas las funciones en un espacio en perspectiva.

•Operaciones Boleanas en 3D.

•Sonido 3D.

•Ayuda interactiva.

•Bookmarks en 3D.

•Amplias opciones de importación y exportación de objetos.

•Soporte para GZIP y proxys.

Características únicas en la versión Pioneer Pro:

•Deformaciones orgánicas.

•Dimensiones en unidades reales.

•Reducción automática del número de polígonos.

•Edición contextual, de puntos, vértices y caras.

•Varias vistas ortogonales o en 3D.

•Vistas múltiples: es posible abrir cuatro vistas adicionales ya sean estas perspectivas o cámaras.

•Características del modelado: permite la deformación orgánica y la realización de operaciones boleanas.

•Pioneer Pro incorpora también una herramienta de dimensionado, que permite a los desarrolladores crear escenas basadas en dimensiones reales. De este modo, se puede reproducir elementos arquitectónicos y objetos con una gran precisión de escalado. Dispone también de una ventana de información sobre el objeto que permite consultar rápidamente los datos relativos a este: tamaño en cada eje, número de vértices, situación, rotación, etc.

Con la herramienta "full axis control" el usuario podrá modificar y ajustar los ejes del objeto con gran simplicidad, lo que unido a la posibilidad de aumentar la resolución con la que visionamos el objeto que estamos creando, conseguiremos resultados profesionales.





Una vez que los objetos están creados, llega el momento de darles la textura y el color. Con este programa, podremos ver el efecto de cada textura en tiempo real, simplemente deberemos seleccionar una entre las distintas posibilidades que ofrece el programa y aplicarla al objeto que estamos creando. Para la aplicación de las texturas podremos escoger entre utilizar la misma para todas las caras del objeto o combinarlas.

•Flujo de navegación automático: A la hora de construir un mundo virtual se ha de tener muy presente que la capacidad de las líneas de co-





nexión a Internet no disponen de la velocidad de transmisión necesaria para no convertir la operación de recibir un fichero de VRML en algo tedioso.







La experiencia de esperar ansioso la recepción de la información, sin que acabe nunca de recibirse es, por propia experiencia, bastante desalentadora.

Pensando en esta situación, los creadores de Caligari Pioneer Pro lo han dotado de un sistema para reducir automáticamente los polígonos, y crear objetos con nivel de detalle, con el fin de dinamizar la navegación a través del espacio virtual.

•Sonido:

Si deseamos añadir mayor realismo a nuestro mundo, una vez terminado el proceso de modelado, podemos añadir sonidos. Pioneer Pro es la primera herramienta de autor que incorpora "Intel 3-D Sound".

Mediante el sonido en 3D, los usuarios dispondrán de un mayor grado de interactividad, puesto que este tipo de sonido, crea una relación "espacial" entre el mundo y sus navegantes. De este modo, asociando ficheros de sonido a distintos objetos, los usuarios podrán escuchar mayor volumen cuando se aproximen a ellos.

Los enlaces con Caligari Pioneer son extremadamente fáciles de crear.

Propiciando que con un simple clic del ratón se tenga la posibilidad de crear enlaces usando las URL, ya sean bien en 2D o bien en 3D. Lo que nos posibilita, por ejemplo, la creación de una tienda virtual en 3D, y que cada producto representado en el mundo virtual, enlace una página web con la descripción de éste.